

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-157177

(43)Date of publication of application : 30.05.2003

(51)Int.Cl.

G06F 9/46

(21)Application number : 2001-357509

(71)Applicant : HITACHI LTD

(22)Date of filing : 22.11.2001

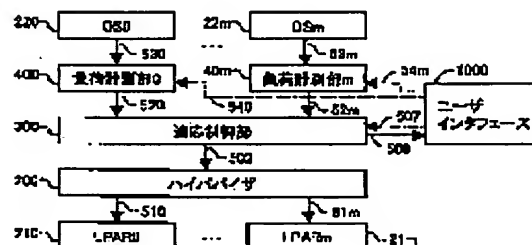
(72)Inventor : KAWAMOTO SHINICHI
HIGUCHI TATSUO
HAMANAKA NAOKI

(54) VIRTUAL COMPUTER SYSTEM AND PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a virtual computer system and a method, which automatically and most suitably assign computer resources to LPARs on the basis of the load of an OS on each LPAR of the virtual computer system and setting information based on knowledge of the workload operated in each OS.

SOLUTION: A load measuring part is mounted on each OS to measure the load of the OS, and knowledge related to the workload of each OS is set through a user interface by an administrator. An adaptive control part obtains an assignment rate of computer resources to each LPAR in accordance with values of the load and setting and gives an assignment change indication to a hypervisor to change assignment.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2003-157177
(P2003-157177A)

(43) 公開日 平成15年5月30日 (2003.5.30)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
G 0 6 F 9/46	3 5 0 3 4 0	G 0 6 F 9/46	3 5 0 5 B 0 9 8 3 4 0 D

審査請求 未請求 請求項の数15 O L (全 19 頁)

(21) 出願番号 特願2001-357509(P2001-357509)

(22) 出願日 平成13年11月22日 (2001.11.22)

(71) 出願人 000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目6番地
(72) 発明者 川本 真一
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内
(72) 発明者 樋口 達雄
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内
(74) 代理人 100075513
弁理士 後藤 政喜 (外2名)

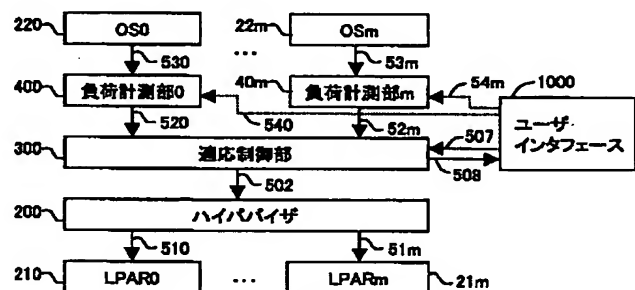
最終頁に続く

(54) 【発明の名称】 仮想計算機システム及びプログラム

(57) 【要約】

【課題】 仮想計算機システムの各L P A R上のOSの負荷と、各OSで動作するワークロードの知識に基づく設定情報を元に、各L P A Rに対する計算機資源の割当を自動的かつ最適に行う仮想計算機システムおよび方法を提供する。

【解決手段】 各OS上に負荷計測部を搭載してOS負荷を計測し、各OSのワークロードに関する知識は、管理者がユーザインタフェースから設定する。適応制御部は、負荷と設定の値に従って各L P A Rに対する計算機資源の割当率を求め、ハイパバイザに対して割当変更指示を出して割当変更する。



【特許請求の範囲】

【請求項1】 物理計算機を複数の論理区画に分割し、各論理区画上でそれぞれOSを動作させ、各論理区画に対する物理計算機の資源の割当を制御するハイババイザを有する仮想計算機システムであって、前記仮想計算機システムの制御動作に拘わる一つまたは複数の設定を入力するユーザインタフェースと、該ユーザインタフェースから入力された設定に従って、各論理区画上のOSの負荷を計測する負荷計測手段と、前記ユーザインタフェースから入力された設定と、該負荷計測手段によって計測された各論理区画上のOSの負荷に基づいて、各論理区画に割当てた計算機資源の割当率を決定し、該割当率が前回の割当率と異なる場合は前記ハイババイザに対し割当率の変更を指示する適応制御手段とを有し、

前記ハイババイザは、該適応制御手段からの指示に従って各論理区画に対する計算機資源の割当率を動的に変更する割当率変更手段を設けたことを特徴とする仮想計算機システム。

【請求項2】 前記ユーザインタフェースは、前記制御動作に拘わる設定として、負荷の種類を指定する負荷種類設定手段を有し、

前記負荷計測手段は、該負荷種類設定手段で指定された種類の負荷を計測し、

前記適応制御手段は、該負荷計測手段によって計測された負荷に基づいて各論理区画に対する計算機資源の割当率を決定することを特徴とする請求項1に記載の仮想計算機システム。

【請求項3】 前記負荷種類設定手段で設定可能な負荷の種類は、CPU使用率、メモリ使用率、ディスク使用率、ネットワーク使用率の中の少なくとも一つを含み、該負荷種類の中から一つ、または、複数の負荷を選択して指定することを特徴とする請求項2に記載の仮想計算機システム。

【請求項4】 前記ユーザインタフェースは、前記制御動作に拘わる設定として、制御の時間間隔を指定する制御インターバル設定手段を有し、

前記負荷計測手段は、該制御インターバル設定手段で指定された制御の時間間隔毎に各論理区画上のOSの負荷を繰り返し計測し、

前記適応制御手段は、該ユーザインタフェースで指定された制御の時間間隔毎に、割当率決定と割当率変更の指示を繰り返して行うことを特徴とする請求項1に記載の仮想計算機システム。

【請求項5】 前記ユーザインタフェースは、前記制御動作に拘わる設定として、適応制御手段の機能を有効または無効のいずれかに指定する適応制御設定手段を有し、前記負荷計測手段は該適応制御設定手段によって適応制御手段の機能が有効に指定された場合にのみ、各論理区画上のOSの負荷を計測し、

前記適応制御手段は前記適応制御設定手段が適応制御手段の機能を有効に指定した場合にのみ、割当率決定と割当率変更の指示を行うことを特徴とする請求項1に記載の仮想計算機システム。

【請求項6】 前記ユーザインタフェースは、前記制御動作に拘わる設定として、前記負荷計測手段で計測した負荷に対する加工の有無及び加工の種類を指定する負荷加工設定手段を有し、

該負荷加工設定手段で負荷に対して加工を施す設定が指定された場合、

前記負荷計測手段は各論理区画上のOSの負荷を計測し、

計測した負荷に対して前記負荷計測手段または適応制御手段は負荷加工設定手段で指定された加工を施し、

前記適応制御手段は、該加工された負荷に基づいて各論理区画に割当てた計算機資源の割当率を決定し、割当率変更の指示を行う一方、

前記負荷加工設定手段で負荷に対し加工を施さない設定が指定された場合、

前記負荷計測手段は各論理区画上のOSの負荷を計測し、

前記適応制御手段は該負荷の値に基づいて割当率決定と割当率変更の指示を行うことを特徴とする請求項1に記載の仮想計算機システム。

【請求項7】 前記負荷加工設定手段は、移動平均または規格化を指定可能であって、該負荷加工設定手段で移動平均が指定された場合では、前記負荷計測手段または適応制御手段は負荷計測手段が計測した負荷に対する加工として、最新の負荷の値を含む指定された個数の負荷の移動平均を取って、得られた値を加工負荷とする一方、前記負荷加工設定手段で規格化が指定された場合では、前記負荷計測手段または適応制御手段は負荷計測手段が計測した負荷に対する加工として、最新の負荷に規格化を施すことを特徴とする請求項6に記載の仮想計算機システム。

【請求項8】 前記ユーザインタフェースは、前記制御動作に拘わる設定として、前記適応制御手段が負荷に基づいて各論理区画に対する計算機資源の割当率を計算する方法を指定する割当率計算方法指定手段を有し、前記適応制御手段は、該割当率計算方法指定手段で指定された割当率計算法に従って割当率を決定し、割当率変更の指示を行うことを特徴とする請求項1に記載の仮想計算機システム。

【請求項9】 前記割当率計算方法指定手段は、比例法または閾値法を指定可能であって、

該割当率計算方法指定手段にて比例法が設定された場合、

前記適応制御手段は負荷計測手段が計測した各論理区画上のOSの負荷に比例して各論理区画に割当てた計算機資源の割当率を決定して、割当率変更の指示を行う一

方、
前記割当率計算方法指定手段にて閾値法が設定された場合、
前記適応制御手段は負荷計測手段が計測した各論理区画上のOSの負荷のいずれかが高負荷判定閾値を超えたら、当該論理区画以外の論理区画に割当てる計算機資源の割当率を減らし、減らした分の合計資源率を当該論理区画の割当率に加えるように割当率を決定して、割当率変更の指示を行い、該高負荷状態の論理区画上のOSの負荷が低負荷判定閾値より小さくなれば、各論理区画に割当てる計算機資源の割当率を変更以前の値に復帰させ、割当率変更の指示を行うことを特徴とする請求項8に記載の仮想計算機システム。

【請求項10】前記ユーザインタフェースは、前記制御動作に拘わる設定として、各論理区画毎に割当てる計算機資源の割当量または割当率の最小値と最大値を指定する割当範囲設定手段を有し、
前記適応制御手段は負荷計測手段が計測した各論理区画上のOSの負荷の値に基づいて各論理区画に割当てる計算機資源を決定する際に、割当率または割当量が前記割当範囲設定手段で指定された最小値以上最大値以下となるように割当率を調整し、割当率変更指示を行うことを特徴とする請求項1に記載の仮想計算機システム。

【請求項11】前記ハイパバイザは、契約を結んだ顧客毎に論理区画を割当て、
顧客との契約条件に応じて該顧客に対応した論理区画に割当てる計算機資源の割当率の最小値と最大値を設定する手段とを有することを特徴とする請求項10に記載の仮想計算機システム。

【請求項12】前記ハイパバイザは、契約を結んだ顧客毎に論理区画を割当て、かつ、前記ハイパバイザに各顧客毎に契約条件を設定する契約ユーザインタフェースを設け、
顧客が該契約ユーザインタフェースにより契約条件を指定すると、契約ユーザインタフェースは設定された該契約条件に応じて、当該顧客に対応した論理区画に割当てる計算機資源の割当率の最小値と最大値を設定することを特徴とする請求項10に記載の仮想計算機システム。

【請求項13】物理計算機を複数の論理区画に分割し、各論理区画上でそれぞれOSを動作させ、各論理区画に対する物理計算機の資源の割当を制御するハイパバイザを有する仮想計算機システムであって、
各論理区画上のOSの負荷を計測する負荷計測手段と、
該負荷計測手段によって計測された各論理区画上のOSの負荷に基づいて、各論理区画に割当てる計算機資源の割当率を決定し、該割当率が前回の割当率と異なる場合は前記ハイパバイザに対して割当率の変更を指示する適応制御手段と、
前記負荷計測手段が計測した負荷、及び該適応制御手段が決定した各論理区画に対する計算機資源の割当率の少

なくとも一方を出力する出力ユーザインタフェースとを有し、

前記ハイパバイザは前記適応制御手段からの指示に従って各論理区画に対する計算機資源の割当率を動的に変更する割当率変更手段を設けたことを特徴とする仮想計算機システム。

【請求項14】ハイパバイザによって物理計算機を複数の論理区画に分割し、各論理区画上でOSを動作させる仮想計算機のプログラムであって、
各論理区画上で動作するOSの負荷を計測する手段と、
該負荷に基づいて各論理区画に割当てる物理計算機資源の割当率を決定する手段と、
該割当率が前回の割当率と異なる場合に、各論理区画に対する計算機資源の割当率が該割当率となるように割当率を変更する手段とを物理計算機に機能させることを特徴とするプログラム。

【請求項15】制御に拘わる設定を指定するユーザインタフェースを有し、ハイパバイザによって物理計算機を複数の論理区画に分割し、各論理区画上でOSを動作させる仮想計算機のプログラムであって、
前記ユーザインタフェースで指定された設定に従って各論理区画上で動作するOSの負荷を計測する手段と、
該負荷に基づいて各論理区画に割当てる物理計算機資源の割当率を決定する手段と、
該割当率がこれまで割当ててきた割当率と異なる場合に、各論理区画に対する計算機資源の割当率が該割当率となるように割当率を変更する手段とを物理計算機に機能させることを特徴とするプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は仮想計算機システムに関し、各L P A R上のOS上で処理しているワークロードに関する少量の知識と、各OSの負荷に従って、計算機資源の各L P A Rに対する割当を自動的に動的変更する技術に関する。

【0002】

【従来の技術】仮想計算機システムは、ハイパバイザにより、物理計算機を複数の論理区画（L P A R : L o g i c a l P A R t i t i o n）に分割し、各L P A Rに対して計算機資源（C P U、主記憶、I/O）を割当て、各L P A R上でそれぞれOSを動作させるものである。

【0003】昨今のWeb（World Wide Webの略称）を利用した計算機システムに対するアクセスは、一般に負荷の予測が困難で、突然アクセスが集中して負荷ピークが現れることがしばしばある。またピーク以外の平常時においては一般に負荷が低いことが知られている。

【0004】時々訪れる負荷ピークのために、初めから多くの計算機資源をL P A Rに対して割当てておくのではなく、平常時には少しの資源を割当てておき、負荷ピ

ークが来たらそれに適応して割当てる資源を増やすことで（これを負荷適応制御と呼ぶ）、無駄な計算機資源を削減したり、あるいはサポートできるL P A Rの数を増やすことができる。

【0005】これを実現するには、まず各L P A Rに対する計算機資源の割当を動的に変更する必要がある。参考文献「H I T A C プロセッサ資源分割管理機構（P R M F）（日立製作所マニュアル8080-2-148-40）」には、各L P A Rに対する計算機資源の割当てを動的に変更する記述がある。それによると、各L P A Rに対する計算機資源の割当を変更する場合、オペレータ（管理者）が操作をして資源割当変更命令を発行し、この命令に従ってハイパバイザは各L P A Rに対する計算機資源の割当を動的に変更する。

【0006】このようなオペレータ操作に基づく割当変更は、システムダウンなどの緊急時や突然の負荷ピークなど迅速に割当変更を必要とする場合には対応できなかった。

【0007】これに対して、特開平9-26889号公報では外部条件の変化に応じて自動的にC P U割当量を変更する仮想計算機システムを開示している。この発明では、緊急事態や運用スケジュールに従って、各L P A Rに対する計算機資源の割り当てをオペレータの介入無しに自動的に変更できる。また、C P U割当量の定義値と実際のプロセッサ使用時間とを比較することにより、プロセッサ使用時間の過不足に応じてプロセッサ割当率の定義値を変更できる。

【0008】

【発明が解決しようとする課題】しかしながら、上記従来の発明においては、プロセッサ使用時間の過不足に応じた割当を行っているが、C P Uの使用時間で計算機システムの負荷を知ることは困難であり、従って負荷に適応して計算機資源の各L P A Rに対する割当率を適切に変更することはできなかった。

【0009】また、各L P A Rの負荷の値を正しく知ることができたとしても、負荷のみから各L P A Rに対する計算機資源の適切な割当率を常に正しく計算することは困難であり、特に、各L P A R上のO S上で実行されるワークロード性質（定常時の負荷、ピーク時の負荷、ピーク幅など）が異なれば、各L P A Rに対する計算機資源の適切な割当率は異なると考えられるからである。

【0010】正しい負荷の値とワークロードに関する少量の知識を合わせることで、各L P A Rに対する計算機資源の割当を自動的にかつ適切に行うシステムを提供できると考えられる。

【0011】そこで本発明の課題は、各L P A R上で動作するO Sの負荷に適応して、また、管理者がワークロードに関する少量の知識（ワークロードの性質）をシステムの制御のパラメータとして与えることにより、各L P A Rに対する計算機資源の割当量を自動的にかつ適切

に行う仮想計算機システム及びプログラムを提供することにある。

【0012】

【課題を解決するための手段】本発明は、物理計算機を複数のL P A R（論理区画）に分割し、各L P A R上でそれぞれO Sが動作し、各L P A Rに対する物理計算機の資源の割当を制御するハイパバイザ（割当手段）を有する仮想計算機システムであって、仮想計算機システムの制御動作に拘わる一つまたは複数の設定を入力するユーザインタフェースと、該ユーザインタフェースから入力された設定に従って、各L P A R上のO Sの負荷を計測する負荷計測手段と、該ユーザインタフェースから入力された設定と、該負荷計測手段によって計測された各L P A R上のO Sの負荷に基づいて、各L P A Rに割当てる計算機資源の割当率を決定し、該割当率がこれまで割り当ててきた割当率と異なる場合はハイパバイザに対し割当率の変更を指示する（割当率変更）適応制御手段を有し、ハイパバイザは該適応制御手段からの指示に従って各L P A Rに対する計算機資源の割当率を動的に変更する手段を設ける。

【0013】

【発明の効果】したがって本発明は、各L P A R上で動作する各O Sの負荷と、各O S上で動作するワークロードに関する知識に基づいた設定情報から、各L P A Rに対して計算機資源を動的にかつ最適に配分し、管理が容易で顧客との契約に合わせた性能を保証できる仮想計算機システムを提供することが可能となり、あるいは、そのような仮想計算機システムにおいて資源を最適に配分するプログラムを提供することが可能となる。

【0014】

【発明の実施の形態】以下、本発明の一実施形態を添付図面に基いて説明する。

【1. 物理計算機】図1に、本発明の仮想計算機システムを動作させる物理計算機130の構成を示す。100～10nはC P U0～C P Unを、120～12kはI/O0～I/Okを示す。111は主記憶を表し、110はC P U（100～10n）とI/O（120～12k）を主記憶111と結合するメモリコントローラを示す。

【0015】なお、C P Uは1台でも良いし、2台以上であっても良い。C P Uが2台以上の場合、各C P U（110～10n）は主記憶111を共有する密結合型マルチプロセッサであるとする。

【0016】140は本物理計算機のコンソールでありI/O0（120）に接続されている。

【2. 仮想計算機システム】図2に本発明を構成する仮想計算機システムの階層図を示す。

【0017】物理計算機130上でハイパバイザ200を動作させる。ハイパバイザは物理計算機130を2つ以上の論理区画（L P A R : L o g i c a l P A R t

ition) LPAR0 (210) ~ LPARm (21m) に分割する。LPAR0 ~ LPARm のそれぞれで OS0 (220) ~ OSm (22m) を動作させ、各 OS 上でそれぞれアプリケーション0 (230) ~ アプリケーションm (23m) を動作させる。

【0018】ハイバイザは、各 LPAR (210 ~ 21m) に対して物理計算機130の CPU (100 ~ 10n)、主記憶111、I/O (120 ~ 12k) (計算機資源と呼ぶ) を割当てる。

【3. 専用割当と共用割当】ハイバイザが計算機資源を各 LPAR に割当てる方法は、専用割当と共用割当の二種類ある。

【0019】専用割当は、特定の計算機資源を特定の LPAR に専用的に割当てる方法である。計算機資源のうち、主記憶 (111) と I/O (120 ~ 12k) は共用割当される。

【0020】なお、CPU (100 ~ 10n) を専用割当にすることもできる。CPU の専用割当の場合、ある LPAR に対して専用割当する CPU の数をその LPAR に対する CPU の割当量と呼ぶ。

【0021】一方、共用割当は、計算機資源を各 LPAR に少しづつ時分割によって割り当てる。この共用割当は、CPU に対してのみ行われる。ある LPAR に対し CPU を割当てている時間の、全 LPAR に CPU を割当てている時間に対する割合を、CPU 割当率と呼ぶ (% で表す。値は 0 ~ 100 の間)。

【0022】このように、専用割当は量を単位とするが、共用割当は率を単位とする。しかし、専用割当において、ある LPAR に専用割当する CPU 数の全 CPU 数 $k+1$ に対する割合を CPU 割当率 (% で表現する。値は 0 ~ 100 の間) とすれば、専用割当も共用割当も同様に率を単位として割当を指示することができる。

【0023】例えば、2つの CPU (CPU0 と CPU1) を持つ物理計算機を2つの LPAR (LPAR0 と LPAR1) に分割して使用する場合、LPAR0 と LPAR1 に対する CPU の割当率をそれぞれ 50 % とすると、空間分割では各 LPAR に対し CPU が一つづつ割当てられ、時間分割では2つの CPU を同一の時間 (タイムスライス) だけ LPAR0 と LPAR1 に交互に割当てるという意味になり、どちらの場合にも適用できる。

【0024】ただし、空間分割では割当率は CPU の台数によって規格化されてしまうが (例えば 2 CPU の場合の割当率は 0 %、50 %、100 % のいずれか)、時間分割にはそのような制約はなく、割当率を 0 ~ 100 % の範囲で自由に指定できる。

【0025】従って以後の説明では、割当率に制約の無い時間分割を割当方法として考えるが、空間分割の場合も割当率に制約を設けることによって適用できる。

【0026】また、以下の説明では各 LPAR に対して動的に配分する計算機資源は、CPU (100 ~ 10

n) のみであるとする。主記憶111やI/O装置 (120 ~ 12k) に対する動的配分も CPU の場合と同様である。

【4. 計算機資源の動的割当変更】ハイバイザ200は予めシステム運用前に設定された各 LPAR (210 ~ 21m) に対する計算機資源の割当率に従って、各 LPAR に対して計算機資源を割当てる。

【0027】管理者 (オペレータ) が物理計算機のコンソール140から割当率を変更すると、ハイバイザ200は各 LPAR (210 ~ 21m) に対する計算機資源の割当率を変更する。

【0028】あるいは、ハイバイザ内に設定されたスケジュールに従い、設定された時刻が来ると割当率を変更する。

【0029】これらの変更方法に加えて、本発明では各 LPAR (210 ~ 21m) 上の OS (220 ~ 22m) の上で動作するアプリケーションからハイバイザに対して資源割当変更命令を発行し、資源割当変更命令を受けたハイバイザは即座にその要求に答えて割当変更を行う機能を設ける。

【0030】この機能は、OS 上のアプリケーションから、ハイバイザのコードをフックする仕組みを実装することによって、容易に実現することができる。

【5. 構成】図3に本発明を適用した仮想計算機システムの機能モジュールの構成を示す。

【0031】本発明の仮想計算機システムは、上記図2の一部に示した仮想計算機の機能モジュールに、負荷計測部0 (400) ~ 負荷計測部m (40m) と、適応制御部300、および、ユーザインタフェース1000を加えた構成になっている。

【5. 1 負荷計測部】負荷計測部0 (400) ~ 負荷計測部m (40m) は、LPAR0 (200) ~ LPARm (21m) 上で動作する OS0 (220) ~ OSm (22m) の上で動作するアプリケーションであり、OS0 (220) ~ OSm (22m) の負荷を計測する。

【0032】負荷計測部0 (400) ~ 負荷計測部m (40m) は OS0 (220) ~ OSm (22m) の負荷計測ライブラリを呼び出し、CPU 使用率、メモリ使用率、ディスク使用率 (ビジー率)、ネットワーク使用率 (ビジー率) などの負荷を取得 (530 ~ 53m) する。

【0033】これら負荷の値は % を単位とし、範囲は 0 ~ 100 とする。負荷計測部0 (400) ~ 負荷計測部m (40m) は、後述するユーザインタフェース1000から計測すべき負荷の種類や制御インターバルの設定情報を受け取り (540 ~ 54m)、その設定情報に基づいて負荷を計測する。負荷計測部0 ~ 負荷計測部m がそれぞれ計測した OS0 ~ OSm の負荷 L0 ~ Lm は適応制御部300に送られる (520 ~ 52m)。

【5. 2 適応制御部】適応制御部300は、OS 上のア

アプリケーションとしてOS0(220)~OSm(22m)のいずれかに搭載する。適応制御部300は520~52mで負荷計測部0(400)~負荷計測部m(40m)のそれぞれからOS0(220)~OSm(22m)の負荷L0~Lmを受取り、LPAR0(210)~LPARm(21m)に対する計算機資源の割当率を求め、割当率が前回の割当率と異なる場合は、ハイババイザ200に対し資源割当変更命令502を発行する。

【0034】各負荷計測部(400~40m)と適応制御部300との間のやり取りは、ソケット通信等の公知技術を用いる。あるいは、これも公知であるが、ハイババイザ200を経由して各LPAR0(210)~LPARm(21m)上のOS0(220)~OSm(22m)同士が通信するLPAR間通信技術を用いても良い。

【0035】適応制御部300は、後述するユーザインタフェース1000から計算機資源の割当率の決定方法に関する様々な設定情報を受け取り(507)、この設定情報に従い計測負荷L0~Lmから各LPAR(210~21m)に対するCPUの割当率S0~Smを求める。なお、各LPAR0~LPARmに対するCPU割当率S0~Smの合計S0+S1+...+Smは、100%となる。

【0036】適応制御部300から資源割当変更命令502を受けたハイババイザは、LPAR0(210)~LPARm(21m)に対する計算機資源の割当率をそれぞれS0~Smに変更する(510~51m)。

{5.3 ユーザインタフェース} ユーザインタフェース1000は、管理者やユーザが本発明の仮想計算機システムの負荷計測やCPUの割当率の判定に関する様々な設定を指示するためのインタフェース機能と、各OS(220~22m)の負荷や各LPAR(210~21m)に割当てるCPUの割当率を管理者やユーザに対して表示するインタフェース機能を備える。

【0037】設定情報はユーザインタフェース1000で入力され、各負荷計測部(400~40m)と適応制御部300に渡される(540~54m、507)。また、負荷や割当率の情報508は適応制御部300からユーザインタフェース1000に渡され表示される。

【0038】ユーザインタフェース1000は、OS0(220)~OSm(22m)のいずれかの上に搭載する。ユーザインタフェースの入力画面や出力画面(後述)は、ユーザインタフェース1000を搭載したOSの画面上に表示される。ユーザインタフェース1000と負荷計測部(400~40m)や適応制御部300とのやり取りは、前述のソケット通信やLPAR間通信技術を用いれば良い。

{5.4 入力ユーザインタフェース} 図3のユーザインタフェース1000の機能のうち、各種の設定を指定する入力ユーザインタフェース1001の画面イメージを

図4に示す。

【0039】第1入力項目(適応制御有効設定)1600は負荷適応制御を行うか(適応制御On)行わないかの設定である。1601と1602は二者択一のラジオボタンになっている。負荷適応制御を行う場合は1601を選択し、行わない場合は1602を選択する。

【0040】第2入力項目1500は制御インターバルの設定である。入力欄1501に制御インターバルの値を入力する。単位は秒であるが、ミリ秒や分などでも良い。入力欄1501に入力された制御インターバルは、負荷計測部(400~40m)での負荷計測の間隔、および、適応制御部300での割当率判定の間隔として用いられる。

【0041】第3入力項目(計測負荷設定)1400は、負荷計測部(400~40m)が計測すべきOS(220~22m)の負荷の種類を示す。1401~1404は四者択一のラジオボタンになっており、選択されたものが計測の対象となる。1401はCPU使用率を示し、初期設定ではCPU使用率を計測する。チェックボックス1402、1403、1404はそれぞれ、メモリ使用率、ディスク使用率、ネットワーク使用率を示し、選択したチェックボックスには「レ」の印が表示される。

【0042】第4入力項目(負荷加工設定)1300は、負荷計測部(400~40m)が計測した負荷L0~Lmに対し、どのような加工を施すかの設定を示す。1301、1302、1303は三者択一のラジオボタンになっており、1301を選択すると負荷L0~Lmに対し加工を施さない。1302を選択すると、負荷L0~Lmに対し移動平均を施す。移動平均のサンプル数は1304に指定する。入力欄1304はラジオボタン1302を選択した場合にのみ有効となる。また、ラジオボタン1303を選択すると、負荷L0~Lmに対し規格化を施す。規格化の階数を入力欄1305に指定する。入力欄1305はラジオボタン1303が選択された場合にのみ有効となる。

【0043】以後負荷L0~Lmに対し加工を施して得られた加工負荷を、LA0~LA mと表す。ラジオボタン1301を選択して負荷に対し加工を施さない場合は、加工負荷は計測した負荷と同じ値とする(LA0=L0、...、LA m=Lm)。移動平均と規格化については後述の(負荷加工処理)で説明する。

【0044】第5入力項目(割当率計算設定)1200は、加工負荷LA0~LA mの値から各LPAR(210~21m)に対するCPU割当率を求める割当率計算の方法の設定を示す。1201と1202は二者択一のラジオボタンになっている。ラジオボタン1201を選択すると、割当率計算法として比例法を用いる。

【0045】ラジオボタン1202を選択すると割当率計算法として閾値法を用いる。閾値法を用いる際の高負

荷判定閾値を入力欄 1203 に、低負荷判定閾値を入力欄 1204 に指定する。これら入力欄 1203、1204 は割当率計算法として閾値法が選択された場合にのみ有効となる。

【0046】割当率計算によって求めた各 LPAR (210~21m) に対する CPU 割当率 (仮 CPU 割当率) を SN0~SNm と表す。比例法と閾値法については後述の (割当率計算処理) で説明する。

【0047】第 6 入力項目 (割当率範囲設定) 1100 は、各 LPAR (210~21m) に対する CPU 割当率の範囲 (上限と下限または最大値と最小値) の設定を示す。LPAR 毎に割当率の上限 (1110~111m) と下限 (1120~112m) を指定する。上限、下限とも 0 以上 100 以下の値を取る。割当率計算で求められた各 LPAR (210~21m) に対する仮 CPU 割当率 (SN0~SNm) は、各上限 (1110~111m) 値を超えず下限値 (1120~112m) を下らないように修正される。

【0048】仮 CPU 割当率を修正したものを CPU 割当率 S0~Sm とする。この上限、下限を設定することにより、割当率の最低限の保証や割当率の最大値の制限を制御できる。本値は主に各 LPAR を用いる顧客との契約によって定められる。

【0049】設定した割当率範囲に基づく割当率の修正処理に関しては後述の (割当率修正処理) で説明する。

【0050】1700 は 1100~1600 で指定した設定を実際に有効にするためのボタンである。例えば、第 1 入力項目 1600 で負荷適応制御を行う (適応制御 On) には、まず 1601 を選択し、続いて 1700 を押して設定を有効にする。

【0051】各入力項目 1300、1400、1500 にそれぞれ指定した設定は、負荷計測部 (400~40m) に渡される。また 1100~1600 に指定した全ての設定は適応制御部 300 にも渡される。負荷計測部 (400~40m) と適応制御部 300 はこれらの各種設定に従って処理を行う。

【0052】管理者は各 OS (220~22m) 上で動作するワークロードの性質を考慮して、各入力項目 1100~1600 の各種の項目を設定する。{5. 5 出力ユーザインタフェース} ユーザインタフェース 1000 の機能のうち、各 LPAR (210~21m) の負荷や CPU 割当率を表示する出力ユーザインタフェース 1002 の画面イメージを図 5 に示す。

【0053】表示欄 1800~180m は各 LPAR (210~21m) 毎の負荷の時系列を示す。負荷は各負荷計測部 (400~40m) が計測した負荷 L0~Lm であっても良いし、負荷 L0~Lm に対し加工を施した加工負荷 LA0~LAM であっても良い。その両方を同時に表示しても良い。どちらの負荷を表示するかユーザが指定できるようになっていても良い。

【0054】表示欄 1810 は各 LPAR (210~21m) に対する CPU 割当率の時系列を示す。全ての LPAR (210~21m) の CPU 割当率を合計すると 100% となる。ある時刻における LPAR i に対する CPU 割当率は、1810 のグラフ内の LPAR i に相当する部分の縦方向の長さで表される。

【0055】表示欄 1820 は割当変更が発生した時刻と、割当変更の理由を逐一表示する。

【0056】出力ユーザインタフェース 1002 は、適応制御部 300 から負荷や割当率の情報 508 を取得し、それを上記図 5 のように表示する。

【0057】本発明の仮想計算機を管理する管理者は、出力ユーザインタフェース 1002 を見ることによって、各 LPAR (210~21m) の負荷がどのように変化しているか、適応制御が適切に動作しているかなどの情報を得ることができる。管理者はこの情報を負荷適応制御の設定にフィードバックすることにより、仮想計算機をより効率的に動作させることができる。{5. 6 適応制御処理} 以下では本発明の仮想計算機システムにおける適応制御処理について、図 6 から図 14 のフローチャートを用いて説明する。

【0058】図 6 に負荷適応制御処理の概要を示す。負荷適応制御処理は、まず 2001 において図 4 の入力ユーザインタフェース 1001 で指定された設定を読み込む。設定の読み込みは負荷計測部 (400~40m) および適応制御部 300 で行われる。

【0059】次に 2002 において負荷の計測を行う。負荷計測は負荷計測部 (400~40m) にて行われる。

【0060】次に 2003 において、入力ユーザインタフェースにて指定された負荷適応制御 (1600) を行うかどうかの設定を調べ、負荷適応制御を行う設定になっていれば 2004 以降で負荷適応制御を行い、そうでない場合は処理を終了する。

【0061】2004 では各 LPAR (210~21m) に対する CPU 割当率 S0~Sm を決定し、2005 において、決定した割当率の一つ以上が前回の割当率 S00~S0m と異なる場合、2006 においてハイパバイザ 200 に対し CPU の割当率を変更するように資源割当変更命令を発行して割当を変更し、処理を終了する。割当率が前回の割当率とまったく同じなら、資源割当変更命令を発行せずに処理を終了する。2003~2006 の割当変更指示までの処理は適応制御部 300 にて行われる。

【0062】本発明の仮想計算機システムは、図 6 に示した一連の処理を制御インターバル間隔で繰り返し行う。制御インターバルは入力ユーザインタフェース 1001 の 1501 に指定された値である。

{5. 7 負荷計測処理} 図 6 の適応制御処理の内、負荷計測処理 2002 の詳細を図 7 に示す。負荷計測処理は

負荷計測部(400~40m)毎に行われ、入力ユーザインタフェース1001の計測負荷種類1400で指定された負荷の種類に従って各OS(220~22m)の負荷 $L_0 \sim L_m$ を計測する。

【0063】すなわち、 $LPAR_i$ においては、2007で計測負荷種類がCPU使用率である場合は、2008でCPU使用率を計測し、得られた値を負荷 L_i とする。

【0064】そうでなく、2009で計測負荷種類がメモリ使用率であれば、2010でメモリ使用率を計測し、得られた値を負荷 L_i とする。

【0065】そうでなく、2011で計測負荷種類がディスク使用率であれば、2012でディスク使用率を計測し、得られた値を負荷 L_i とする。

【0066】そうでなく、2013で計測負荷種類がネットワーク使用率であれば、2014でネットワーク使用率を計測し、得られた値を L_i とする。各種の使用率の計測はOS(220~22m)が備える負荷計測ライブラリ等を用いる。ここで、ネットワーク使用率とは、例えば、接続数、リクエスト数などであり、適宜設定されるものである。

〔5. 8 割当率決定処理〕図6の適応制御処理の中の割当率決定処理2004の詳細を図8に示す。割当率決定処理は、まず2020において負荷計測処理によって計測された各OS(220~22m)の負荷 $L_0 \sim L_m$ に対し加工を施し加工負荷 $LA_0 \sim LA_m$ を求める。

【0067】そして2021において、加工負荷 $LA_0 \sim LA_m$ に基づいて、 $LPAR_0(210) \sim LPAR_m(21m)$ に対する仮CPU割当率 $SN_0 \sim SN_m$ を求める(割当率計算)。

【0068】そして2022で、仮CPU割当率 $SN_0 \sim SN_m$ が入力ユーザインタフェース1001によって指定された $LPAR$ 毎の割当率範囲の上限(1110~111m)と下限(1120~112m)の間に収まるように仮CPU割当率 $SN_0 \sim SN_m$ を修正してCPU割当率 $S_0 \sim S_m$ を求め、処理を終了する。

〔5. 9 負荷加工処理〕図8の割当率決定処理の中の負荷加工処理2020の詳細を図9に示す。

【0069】まず、入力ユーザインタフェース1001の1300において負荷加工を施さない設定(無変換)が選択された場合を2030でチェックし、加工を施さない場合は2031において、負荷 $L_0 \sim L_m$ をそのまま加工負荷 $LA_0 \sim LA_m$ とし($LA_0 = L_0$ 、 \dots 、 $LA_m = L_m$)、処理を終了する。

【0070】一方、加工を施す場合は2032において、加工の種類が移動平均法であるかどうかを判定し、移動平均法であれば、2033において、負荷に移動平均を施す。

【0071】移動平均は、負荷計測部(400~40m)が計測した負荷 $L_0 \sim L_m$ の過去の値を保存してお

く。保存する数は、入力ユーザインタフェース1001の負荷加工設定1300の1304に指定されたサンプル数(S と表す)から1を引いた $S-1$ 個である。

【0072】ここで、 k 個前($0 < k < S$)の $LPAR_i$ 上の OS_i の負荷 L_i を $L_i(k)$ と表す。 OS_i 毎に負荷 $L_i(S-1)$ 、 \dots 、 $L_i(1)$ を保存しておくことになる。移動平均はOS毎に最新の負荷 L_i を含めた S 個の負荷系列 $L_i(0)$ 、 $L_i(1)$ 、 \dots 、 $L_i(S-1)$ の平均値を求める。そして求めた値を加工負荷 LA_i とする。すなわち、 $LA_i = (L_i(0) + L_i(1) + \dots + L_i(S-1)) / S$ となる。この計算をすべてのOS($LPAR$)について行い $LA_0 \sim LA_m$ を求め処理を終了する。

【0073】一方、加工方法として規格化法が選択された場合は、2034において、 OS_i の負荷 L_i に対して規格化を施す。

【0074】規格化は、値を予め設定された飛び飛びの値のいずれかに合わせる。規格化の階数は入力ユーザインタフェース1001の負荷加工設定1300の1305(階数)に指定された値(N)とする。

【0075】 OS_i の負荷 L_i に対し、階数 N の規格化を施して得られた加工負荷 LA_i は、 $LA_i = (\text{floor}(L_i \times N / 100) + 1) \times 100 / N$ から求める。この計算をすべてのOS($LPAR$)について行い、加工負荷 $LA_0 \sim LA_m$ を求め処理を終了する。

【0076】なお、負荷の加工はOS(220~22m)毎に負荷計測部(400~40m)で行っても良いし、適応制御部300でまとめて行っても良い。負荷計測部で行う場合は、負荷計測部(400~40m)から適応制御部300に送る値520~52mは加工負荷 $LA_0 \sim LA_m$ である。

【0077】適応制御部300で負荷加工を行う場合は、負荷計測部(400~40m)から適応制御部300に送る値520~52mは負荷 $L_0 \sim L_m$ である。

〔5. 10 割当率計算処理〕図8の割当率決定処理の中の割当率計算処理2021の詳細を図10に示す。

【0078】まず2040において、入力インタフェース1001の割当率計算設定1200で比例法が選択されたか、閾値法が選択されたか調べ、比例法であれば2042~2045の処理を行う。

【0079】(5. 10. 1 比例法の処理) 2042でループカウンタ i を0に初期化し、2043ではループカウンタ i の値が m より大きくなるまで、2044と2045の処理を繰り返し行う。

【0080】2044では、加工負荷 $LA_0 \sim LA_m$ に基づき仮CPU割当率 SN_i を求める。2044の計算で ΣLA_i は、全ての加工負荷 $LA_0 \sim LA_m$ の和を示す。

【0081】 $SN_i := 100 LA_i / \Sigma LA_i$ は、全加工負荷値の和に対する OS_i の加工負荷 LA_i

の割合をパーセントで表したものであり、 $LPAR_i$ に対するCPU割当率 SN_i は、 OS_i の加工負荷 LA_i に比例した値となる。

【0082】2045では、ループカウンタ i を1だけインクリメントし、2043に戻る。2043から2045の一連の処理が繰り返し行われ、 $LPAR_0$ (210)～ $LPAR_m$ (21m)に対する仮CPU割当率 SN_0 ～ SN_m が求まり、処理を終了する。

(5. 10. 2 閾値法の処理) 一方、割当計算方法が閾値法の場合は2041を実行する。2041の詳細は図11のようにになっている。

【0083】まず、2050においてループカウンタ i を0に初期化し、2051においてループカウンタ i が m より大きくなるまで2052以降の処理を繰り返す。2052では、 OS_i の加工負荷 LA_i が入力ユーザインタフェース1001の割当率計算方法設定1200の閾値法の1203に指定された高負荷判定閾値 TH の値より大きく ($LA_i > TH$)、かつ OS_i が高負荷状態にあることを示すフラグ Hi が立っていないならば ($Hi = 0$)、つまりこれまで低負荷状態であったが負荷が上がった場合、2053に進んで OS_i が高負荷状態になった場合の処理を行う。

【0084】すなわち、2053では仮CPU割当率 SN_i を $100 - (100 - SP_i) L_{oi} / B$ とする。ここで、 SP_i は $LPAR_i$ に対する前回の負荷適応制御で求めたCPU割当率を示す。また L_{oi} は、 OS_i 以外の各 OS_j の加工負荷 LA_j の和を示す。 B は低負荷状態の $LPAR$ 上の OS の負荷を高くしていく上限値を示す。

【0085】現在 OS_i 以外の OS の負荷の合計は L_{oi} であり、このとき、 $LPAR_i$ 以外の $LPAR$ に対するCPU割当率は $(100 - SP_i)$ である。

【0086】現在 OS_i の負荷が高いので、 OS_i が動作する $LPAR_i$ 以外の $LPAR$ に対するCPU割当率を削減し、 $LPAR_i$ に対するCPU割当率を増加させたい。

【0087】そこで、 $LPAR_i$ 以外の $LPAR$ に対するCPU割当率を、 $LPAR_i$ 以外の $LPAR$ で動作する OS の合計負荷が B になるように減らし、減らした分を $LPAR_i$ に対するCPU割当率に加える。

【0088】この計算を式で表すと、
 $SN_i := 100 - (100 - SP_i) L_{oi} / B$
 と表される。また、 SN_i を計算すると同時に、 OS_i が高負荷状態であることを示すフラグ Hi を1にセットする。またループカウンタ j を0にセットする。

【0089】2053は高負荷状態になった OS_i に対する仮CPU割当率の計算であったが、2054～2057は OS_i 以外の $LPAR$ においてCPU割当率を減らす計算を示す。

【0090】すなわち、2054では j が m より大きく

なるまで2055～2057の処理を繰り返す。2055では、 $LPAR_i$ に対する仮CPU割当率 SN_i はすでに2053で計算済みなので、 j が i である場合を判定し、 $j = i$ なら2056の処理を行わない。

【0091】2056では、 $LPAR_i$ 以外の $LPAR$ に対する仮CPU割当率 SN_j を計算する。計算方法は、 $LPAR_i$ に対する仮CPU割当率を除いた (100から引いた) 値を $LPAR_i$ 以外の $LPAR$ の数で割ることによって求める。これを式で表すと、
 $SN_j := (100 - SP_i) L_{oi} / (B * m)$
 となる。

【0092】2057ではループカウンタ j を1だけインクリメントして、2054に戻る。2054のループを抜けると、各 $LPAR$ (210～21m) に対する仮CPU割当率 SN_j が全て求まり、処理が終了する。

【0093】一方、2052において加工負荷 LA_i の値が高負荷判定閾値 TH を超えないか、または、高負荷状態を示す Hi フラグが立っている ($Hi = 1$) 場合は、2058以降の処理を行う。

【0094】2058では、加工負荷 LA_i が低負荷判定閾値 TL より小さく、かつ OS_i が高負荷状態であることを示すフラグ Hi が立っている状態、すなわち、これまで高負荷状態であったが負荷が下がった場合、どの $LPAR$ も平等なCPU割当率となるように割当率を計算する。

【0095】つまり、2059でループカウンタ j を0に初期化し、 OS_i が高負荷状態であることを示すフラグを下ろし ($Hi := 0$)、2060でループカウンタ j が m となるまで繰り返し2061を行う。2061では、 $LPAR_j$ の仮CPU割当率 SN_j を $100 / m$ とし、ループカウンタを1インクリメントして2060に戻る。2060の条件判定が真となってループを抜けると、各 $LPAR$ (210～21m) に対する仮CPU割当率 SN_0 ～ SN_m はすべて $100 / m$ となり、処理を終了する。

【0096】また、2058において OS_i の加工負荷 LA_i が、低負荷判定閾値より大きいあるいは OS_i が高負荷状態でない ($Hi = 0$) 場合は、2062でループカウンタ i を1だけインクリメントし、2051に戻る。

【0097】2051で条件が真となると、いずれの OS (220～22m) においても、2052や2058の条件を満足する (真となる) ような負荷の変化はないため、2063以降で $LPAR_j$ に対する仮CPU割当率 SN_j の値を、一回前の適応制御の際に求めた $LPAR_j$ に対するCPU割当率 SP_j の値とする ($SN_j := SP_j$) ことを各 $LPAR$ (210～21m) について行い (2063、2064、2065)、処理を終了する。

{5. 11 割当率修正処理} 図8の割当率決定処理に示

した割当率修正処理 2022 の詳細を図 12、図 13、図 14 に示す。

【0098】まず、図 12 の 2071~2078 において、LPAR_i に対する仮 CPU 割当率 SN_i の値が、入力ユーザインタフェース 1001 の割当率範囲設定 1100 で指定される LPAR_i (211) に対する割当率の上限 MaxSi (1111) と下限 MinSi (1121) の間に入っているかどうか調べ (MinSi ≤ SN_i ≤ MaxSi)、この上限と下限に入っていない場合には、SN_i に対して、

$$\text{MinSi} \leq \text{SN}_i + d_i \leq \text{MaxSi}$$

を満足する最小限の d_i の値を求める。

【0099】すなわち 2071 においてループカウンタ i を 0 に初期化し、2072 において i が m より大きくなるまで 2073~2078 の処理を繰り返し行う。2073 は仮 CPU 割当率 SN_i が MaxSi より大きな値か調べ、もし大きければ 2074 で d_i の値を MaxSi - SN_i から求める (d_i は負の値)。

【0100】もし 2073 で SN_i が MaxSi 以下である場合は、2075 において仮 CPU 割当率 SN_i が MinSi より小さいか調べ、小さい場合は 2076 で d_i の値を MinSi - SN_i から求める (d_i は正の値)。

【0101】2075 において、SN_i が MaxSi 以上である場合は、MinSi ≤ SN_i ≤ MaxSi を満足するから、2077 において d_i を 0 とする。2074、2076、2077 のいずれかによって d_i の値が求まると、2078 においてループカウンタ i の値を 1 インクリメントし、2072 に戻る。

【0102】2072 の条件判定が真となれば、各 LPAR (210~21m) に対する d_i が求まっている。

【0103】求めた d_i の値を、SN_i に加えて CPU 割当率 S_i とすれば ($S_i := \text{SN}_i + d_i$)、いずれの LPAR に対する CPU 割当率 S_i も、MinSi ≤ S_i ≤ MaxSi を満足する。

【0104】しかし、 d_i によって仮 CPU 割当率を増減しているため、 $\sum S_i$ が 100% とならない可能性がある。

【0105】そこで、2079 以降で、 S_i が上限 MaxSi と下限 MinSi の範囲を満足しつつ、 $\sum S_i = 100$ となるように S_i の値を修正する。

【0106】2079 では $\sum d_i$ (d_i を $i=0 \sim i=m$ まで合計した値) が正であるか調べ、正であった場合は、2080 において d_i の値が 0 以下となるような d_i の個数を x_m として図 13 の 2082 以降を実行する。

【0107】2079 で $\sum d_i$ が 0 以下であった場合は、2081 において d_i の値が 0 以上となるような d_i の個数を x_p として図 14 の 2101 以降を実行す

る。

【0108】図 13 の 2082 ではループカウンタ j を 0 に初期化し、2083 で j が m より大きいか、 $\sum d_i$ の値が 0 となるまで、2084~2089 を繰り返し行う。

【0109】CPU 割当率計算処理によって求められた仮 CPU 割当率 SN_i の合計 $\sum \text{SN}_i$ は 100% となるように求められているので、 $\sum d_i$ が正であるということは、 $\sum (\text{SN}_i + d_i) > 100\%$ となる。

【0110】そこで、一部の LPAR に対する CPU 割当率を、より小さな値にするように修正しなければならない。 d_i が 0 より大きな値になっているということは、仮 CPU 割当率 SN_j そのものは MinS_j より小さな値であり、従って SN_j + d_i は MinS_j となるから、この LPAR に対する CPU 割当率を小さくするわけには行かない。

【0111】つまり CPU 割当率の値を小さく修正する対象は、 d_i が 0 以下の場合についてということになる。そこで 2084 において、 d_i の値が 0 以下であるものについてのみ 2085~2089 の処理を行っている。

【0112】さて、割当率をどれだけ小さくするかは、最終的な CPU 割当率の合計が 100% となるようにすれば良い。そこで、割当率が 100% を超える分 $\sum d_j$ を、割当率を修正できる対象の数 x_m で等分した $\sum d_j / x_m$ を修正対象から引いて修正する。ただし、SN_j + $d_i - \sum d_j / x_m$ が逆に MinS_j より小さくなつてはならないので、これを 2085 において判断する。

小さくならない場合は、2086 において新たに、

$$d_j := d_j - \sum d_j / x_m$$

とする。

【0113】一方、SN_j + $d_i - \sum d_j / x_m$ が MinS_j より小さい場合は、2087 において、CPU 割当率が MinS_j となるように、新たに、

$$d_j := \text{MinS}_j - \text{SN}_j$$

とする。 d_j の値を修正すると 2088 において x_m を 1 だけデクリメントし、2089 でループカウンタ j を 1 だけインクリメントして 2083 に戻る。

【0114】2083 の条件判定が真となってループを終了すると、修正された d_j が求まっている。そこで、2090~2092 において LPAR_i に対する最終的な CPU 割当率 S_i を、

$$S_k := \text{SN}_k + d_k$$

より計算して求め、処理を終了する。なお、2092 では、カウンタ k を 1 だけインクリメントする。

【0115】次に、図 14 の 2101 ではループカウンタ j を 0 に初期化し、2102 で j が m より大きいか、 $\sum d_i$ の値が 0 となるまで、2103~2108 を繰り返し行う。

【0116】CPU 割当率計算処理によって求められた

仮CPU割当率 SN_i の合計 $\sum SN_i$ は、100%となるように求められているので、 $\sum d_i$ が負であるということは、 $\sum (SN_i + d_i) < 100\%$ となる。

【0117】そこで、一部のCPU割当率をより大きな値にするように修正しなければならない。 d_i が0より小さな値になっているということは、仮CPU割当率 SN_j そのものは $MaxS_j$ より大きな値であり、従って $SN_j + d_i$ は $MaxS_j$ となるから、このLPARに対するCPU割当率を大きくするわけには行かない。

【0118】つまりCPU割当率の値を大きな値に修正する対象は d_i が0以上の場合についてということになる。

【0119】そこで2103において、 d_i の値が0以上であるものについてのみ2104～2107の処理を行っている。

【0120】さて、割当率をどれだけ大きくするかは、最終的なCPU割当率の合計が100%となるようにすれば良い。そこで、割当率が100%を下回る分 $\sum d_j$ を、割当率を修正できる対象の数 x_p で等分した $\sum d_j / x_p$ を修正対象から引いて（ $\sum d_j / x_p$ は負の値なので）修正する。

【0121】ただし、 $SN_j + d_i - \sum d_j / x_p$ が逆に $MaxS_j$ より大きくなってはならないので、これを2104において判断する。大きくならない場合は、2105において新たに、

$$d_j := d_j - \sum d_j / x_p$$

とする。

【0122】一方大きい場合は、2106において、CPU割当率が $MaxS_j$ となるように、新たに、

$$d_j := MaxS_j - SN_j$$

とする。 d_j の値を修正すると2107において x_p を1だけデクリメントし、2108においてループカウンタ j を1だけインクリメントして2102に戻る。

【0123】2102の条件判定が真となってループを終了すると、修正された d_j が求まっている。そこで、上記2090～2092（図13）においてLPAR i に対する最終的なCPU割当率 Si を、

$$Si := SN_i + d_i$$

より計算して求め、処理を終了する。

〔6. 全体的な作用〕以上の処理により、各LPAR上のOS上で実行されるアプリケーション（サービス、デーモンを含む）のワークロード性質（定常時の負荷、ピーク時の負荷、ピーク幅などの特性）に応じて、入力ユーザインタフェース1001で各LPARの計測負荷種類を適宜選択し、また、適切な制御インターバル1500を設定することにより、ワークロードの急増（ピークの発生）に対して適切な割当率の変更を行うことが可能となつて、各LPARに対する計算機資源の適切な割当の自動化を実現できるのである。

【0124】例えば、LPAR a のOS a 上でWebサ

ーバが稼動し、LPAR b 上のOS b 上でデータベースサーバが稼動している場合、各LPARのCPU使用率が同等であったとしても、負荷が掛かる部分が異なり

（ワークロード特性）、Webサーバでは、負荷の増大はネットワーク使用率の増大が伴い、データベースサーバでは、負荷の増大はディスク使用率（またはキャッシュ用のメモリ使用率）が増大するという特性がある。

【0125】そこで、管理者は、入力ユーザインタフェース1001で、各LPAR i のOS i 上で稼動するアプリケーションのワークロード特性に応じて、計測負荷種類を決定すればよく、上記の例では、Webサーバが稼動するLPAR a ではネットワーク使用率を選択し、データベースサーバが稼動するLPAR b ではディスク使用率を選択することで、負荷の種類と大きさに応じた計算機資源の動的な割当率変更を適切に行うことが可能となるのである。

【0126】特に、Webサーバ等では、負荷のピークが現れる時刻等を予測するのが非常に難しいため、本発明のように、ワークロード特性に応じて適宜負荷計測の種類を選択して適応制御を行うことにより、負荷の増減に応じた計算機資源の割当率変更を自動的かつ適切に行うことが可能となるのである。

【0127】また、入力ユーザインタフェース1001では、計測した負荷の加工について、無変換、移動平均、規格化のいずれかを選択するようにしたので、各LPARのピークの発生状況などに応じたチューニング（最適化）を行うことが可能となる。

【0128】つまり、負荷加工で無変換を選択すれば、計算機資源の動的な割当率変更が、負荷変動に対してリニアに応答可能となり、移動平均を選択した場合には、負荷の微小な変動に対して割当率の変更が頻繁になるのを抑制して、割当率変更に伴うオーバーヘッドを低減できるとともに、制御インターバル及びサンプル数との組み合わせで幅広いチューニング（最適化）を行うことができ、あるいは、規格化を選択した場合では、負荷の微小な変動に対して割当率の頻繁な変更を抑制して、割当率変更に伴うオーバーヘッドを低減でき、制御インターバル及び規格化の階層数の組み合わせに応じて幅広いチューニングを行うことができる。

【0129】さらに、割当変換（割当率計算）1200では、比例法と閾値法を選択可能としたので、負荷変動に対してリニアに応答する必要がある場合では比例法を適用し、負荷の微小な変動に対して割当率の頻繁な変更を抑制したい場合には閾値法を用いることで、頻繁な割当率変更に伴うオーバーヘッドを低減でき、さらに高負荷、低負荷の閾値に応じて幅広いチューニング（最適化）を行うことができる。

【0130】また、出力ユーザインタフェース1002により、各LPAR i 毎の負荷と時刻の関係と、割当率変更の内容を時系列的に表示するようにしたので、負荷

変動に対してどのように計算機資源の割当率を変更されたのかをユーザ（管理者）に知らせることができ、管理者はこの負荷と割当率の履歴に基づいて、入力ユーザインタフェース 1001 で設定する各種パラメータの検討を行い、各 LPAR_i 毎に最適なチューニングを行うことが可能となる。

【0131】図 15 は第 2 の実施形態を示し、前記第 1 実施形態の一部を変更し、ユーザインタフェース 1000 を搭載する LPAR を独立させたものである。

【0132】以下、前記第 1 実施形態との相違点についてのみ説明する。

【0133】図 15 は、前記第 1 実施形態の図 3 に示した LPAR₀ (210) ~ LPAR_m (21m) 以外に管理目的の LPAR (L PAR_x) を設け、この上で OS_x を動作させ、この OS_x 上に適応制御部 300 とユーザインタフェース 1000 を搭載する。

【0134】ユーザインタフェース 1000 の入力画面や出力画面は OS_x の画面上に表示される。また、OS_x 上には負荷計測部は搭載しない。各負荷計測部 (400 ~ 40m) と適応制御部 300 とユーザインタフェース 1000 との間のやり取りは、ソケット通信や LPAR 間通信技術を用いる。その他は前記第 1 実施形態と同様である。

【0135】この例では、適応制御部 300 及びユーザインタフェース 1000 が管理用の LPAR_x (22x) 上で稼動するため、他の LPAR₀ ~ m では LPAR の動的な割当率変更に必要な計算機資源が不要となっており、各 OS₀ ~ m は負荷計測部 400 ~ 40m の処理を除いてアプリケーションの実行に専念でき、各 LPAR の利用効率を向上させることが可能となる。

【0136】図 16 は第 3 の実施形態を示し、前記第 1 実施形態の一部を変更し、適応制御部 300 とユーザインタフェース 1000 をハイパバイザ 200 の内部に設けたものである。

【0137】以下、前記第 1 実施形態との相違点についてのみ説明する。

【0138】図 16 は、前記実施形態 1 の図 3 に示した適応制御部 300 とユーザインタフェース 1000 を、ハイパバイザ 200 の内部に設けたものである。ユーザインタフェース 1000 の入力画面や出力画面は物理計算機 130 のコンソール 140 に表示される。

【0139】各負荷計測部 (400 ~ 40m) と適応制御部 300 やユーザインタフェース 1000 とのやり取りは LPAR 間通信で用いられる共有メモリを介して行う。また、適応制御部 300 とユーザインタフェース 1000 との間のやり取りはハイパバイザ 200 の内部メモリを用いれば良い。その他の構成は前記第 1 実施形態と同様である。

【0140】図 17 は第 4 の実施形態を示し、前記第 1 実施形態のユーザインタフェースの一部を変更したもの

である。

【0141】以下、前記第 1 実施形態との相違点についてのみ説明する。

【0142】図 17 の入力ユーザインタフェース 1003 は、前記第 1 実施形態の図 4 に示した入力ユーザインタフェースに、Save ボタン 1701 と Restore ボタンを加えたもので、その他の構成は、前記第 1 実施形態の図 4 と同様である。

【0143】この入力ユーザインタフェース 1003 は、表示領域の下部に Save ボタン 1701 と Restore ボタン 1702 を付け加えたものである。

【0144】Save ボタン 1701 を押す（クリックする）と、入力項目 1100 ~ 1600 で指定された各種の設定を、予め設定したディスク上の設定保存ファイルに書き出す。Restore ボタン 1702 を押すと、設定を保存している上記設定保存ファイルを読み出し、入力項目 1100 ~ 1600 を保存された時点の設定に復元する。

【0145】これにより、管理者は本発明の仮想計算機システムを起動するたびに、設定を入力ユーザインタフェース 1003 から入力する必要がなくなり、設定保存ファイルを呼び出すだけで保存されていた設定を復元できる。

【0146】図 18 は、第 5 の実施形態を示し、前記第 1 実施形態のユーザインタフェースの出力インタフェース（出力部）を変更して、ログ記録部 1004 としたもので、その他の構成は前記第 1 実施形態と同様である。以下、前記第 1 実施形態との相違点についてのみ説明する。

【0147】図 18 のログ記録部 1004 は、前記第 1 実施形態の OS₀ (210) ~ OS_m (22m) の何れかの上に搭載する。

【0148】ログ記録部 1004 は、適応制御部 300 から各 OS (220 ~ 22m) の負荷 L₀ ~ L_m または加工負荷 LA₀ ~ LA_m と、各 LPAR (210 ~ 21m) に対する CPU 割当率 S₀ ~ S_m および、割当変更がなされた際の変更理由を一定時間毎に受け取り、それらを時系列としてログファイル 1005 に書き出す。

【0149】管理者は、このログファイル 1005 を参照することにより、負荷変動に対してどのように計算機資源の割当率を変更されたのかを知ることができ、管理者はこの負荷と割当率の履歴に基づいて、入力ユーザインタフェース 1001 で設定する各種パラメータの検討を行って、各 LPAR_i 毎に最適なチューニングを行うことが可能となる。

【0150】なお、このログ記録部 1004 を前記第 2 実施形態に適用する場合は、ログ記録部 1004 を適応制御部 300 を搭載した管理 LPAR (L PAR_x) 上に搭載する。

【0151】また、同じく前記第 3 実施形態に適用する

場合は、ハイパバイザ200の内部にログ記録部1004が設けられる。

【0152】図19は、第6の実施形態を示し、前記第1実施形態のユーザインタフェース1000に、各L PARを利用する顧客に対して提供する契約ユーザインタフェースを付加したもので、その他の構成は、前記第1実施形態と同様である。

【0153】契約ユーザインタフェース0(3000)～契約ユーザインタフェースm(300m)は、それぞれ、L PAR0(210)～L PARm(21m)に対応して設けられている。

【0154】本実施形態の仮想計算機においては、契約を結んだ顧客毎にL PARを用意し、顧客はインターネット(またはネットワーク)を介して、その顧客に割当てられたL PARにアクセスして処理を行う。したがって、契約ユーザインタフェース(3000～300m)は契約を結んだ顧客の計算機の画面(表示手段)上に表示される。

【0155】契約ユーザインタフェース0(3000)は、入力データ600と出力データ610によってユーザインタフェース1000と接続されている。また契約ユーザインタフェースm(300m)は、入力データ60mと出力データ61mによってユーザインタフェース1000と接続されている。

【0156】契約ユーザインタフェース(3000～300m)は、顧客が契約内容を更新したり、本発明の仮想計算機システムが当該顧客に対して割当てている計算機資源の割当率等のサービス状況の表示などを実行する。

【0157】契約ユーザインタフェース(3000～300m)は、図20に示す契約入力ユーザインタフェース3100と、図21の契約確認ユーザインタフェース3200から構成される。あるいは、図22の契約出力ユーザインタフェースを含んでも良い。

【0158】次に、図20の契約入力ユーザインタフェース3100について説明する。

【0159】図20の契約入力ユーザインタフェースは、契約顧客が契約の内容を変更するためのインタフェースである。ここでの契約内容とは、契約しているL PARに対するCPU割当率の上限3101と下限3102の入力欄である。

【0160】顧客は自らに割当てられているL PARに対するCPU割当率の上限と下限を、当該L PAR上で行う処理のワークロードに関する知識を利用しながら、契約入力ユーザインタフェース3100の上限3101と下限3102に指定する。

【0161】3103は、図20の3101と3102に入力された割当率の上限と下限(割当率範囲)を有効とする(契約を変更する)ためのボタンである。

【0162】変更ボタン3103が押され(クリックさ

れ)ると、入力欄3101と3102で指定された割当率の上限と下限の情報が、図19の入力データ600を介してユーザインタフェース1000の入力ユーザインタフェース1001に送られる。

【0163】入力ユーザインタフェース1001は、指定された割当率範囲が妥当なものを確認し、妥当であれば、入力ユーザインタフェース1001の割当率範囲設定1100の当該L PARiの上限(111i)と下限(112i)に送られてきた上限と下限の値を設定し、図4に示した更新ボタン1700を押したのと同様に設定を有効にする。

【0164】割当率範囲が妥当でない場合(他の顧客が下限の値を大きく設定し、当該顧客に対して指定された値を下限に設定できない場合は妥当ではないと判断される。)、送られてきた上限と下限は割当率範囲設定1100には反映されない。

【0165】そして、入力ユーザインタフェース1001は、当該顧客の画面に図21の契約確認ユーザインタフェース3200を表示し、先の契約変更が正しく受け付けられたかどうかを顧客に通知する。

【0166】次に、図21の契約確認ユーザインタフェースについて説明する。

【0167】図21の契約確認ユーザインタフェース3200は、顧客が契約入力ユーザインタフェース3100によって指示した契約変更要求が、正しく受け付けられたかどうかの確認を顧客に提示するユーザインタフェースである。3201は契約変更の受け付け状況を示す。この受け付け状況3201は、契約変更要求が正しく受け付けられた場合はAcceptedを表示し、正しく受け付けられない場合はInvalid等を表示する。

【0168】正しく受け付けられるためには、契約入力ユーザインタフェース3100から入力ユーザインタフェース1001に送られた割当率範囲が妥当なものである必要がある。

【0169】契約内容3202には、契約変更が正しく受け付けられた場合には、変更された契約内容が表示され、契約変更が正しく受け付けられなかった場合には、その理由が表示される。

【0170】次に、図22の契約出力ユーザインタフェースについて説明する。

【0171】図22の契約出力ユーザインタフェース3300は、当該顧客に対応したL PARの上で動作するOSの負荷又は加工負荷の時系列(3301)、当該L PARに割当てられている計算機資源の割当率の時系列(3302)、および割当変更が発生した場合の変更理由3303を表示する。前記第1実施形態に示した図5の出力ユーザインタフェース1002とは異なり、ここで表示されるのは、当該顧客が契約して使用しているL PARに関する情報のみであり、他の顧客が契約しているL PARの情報は表示されない。

【0172】以上により、各L P A Rを利用する顧客は、契約ユーザインタフェース3000~300mにより、利用するL P A R毎の負荷と割当率の変化を時系列に確認できるとともに、契約の範囲内で割当率の変更を行うことが可能となり、顧客は契約内容を常時確認できるようになるとともに、負荷と割当率の変化の履歴に基づいて、自らが割当率を変更することができるため、L P A Rの利用者に対するサービスの向上を図ることができる。

【0173】なお、契約出力ユーザインタフェース3300は、画面（コンソール140）に値を表示するユーザインタフェースであるが、これを前記図18に示したログ記録部1004で置き換えても良い。このときログ記録部1004は、顧客が本発明の仮想計算機にインターネットを介してアクセスする際に使用するアクセス装置（計算機等）に搭載し、ログ記録部1004は顧客のアクセス装置にログファイルを出力するようにしてもよい。

【0174】図23は、第7の実施形態を示し、前記第6実施形態の契約入力ユーザインタフェース3100の入力を変更したものであり、その他の構成は前記第6実施形態と同様である。

【0175】図23において、契約入力ユーザインタフェース3400は、前記図20の契約入力ユーザインタフェース3100に比べて、指定する内容が抽象的になっており、L P A Rの利用者の割当率の変更にかかる操作を簡易にするものである。

【0176】すなわち、サービスのレベルをS、A、B、Cから選択するために、図中3401~3404は四者択一のラジオボタンになっており、3401はレベルSの選択を、3402はレベルAの選択を、3403はレベルBの選択を、3404はレベルCの選択を示す。

【0177】レベルSは最も性能重視の契約を示す。レベルCは最も価格重視の契約を示す。レベルAは性能重視だが、レベルSほど価格は高くなく、レベルBは価格重視だがレベルCより性能を重視する契約である。どのレベルがどのようなサービスを提供するかは、契約書などの記述に基づくものである。

【0178】図中3403は契約変更ボタンであり、このボタンを押すと選択されたサービスレベルが入力ユーザインタフェース1001に送られる。入力ユーザインタフェース1001は、サービスレベルが送られてくると、そのサービスレベルを予め定められたチャート（図示省略）等を参照して、計算機資源の割当率の上限と下限に変換し、それらの値が妥当かどうか判断し（他の顧客と契約したサービスレベルが守れなければ、当該顧客の指示したサービスレベルは妥当でないと判断される）、妥当であれば、入力ユーザインタフェース1001の割当率範囲設定1100の当該L P A R iの上限

（111i）と下限（112i）に、チャートを参照して得られた上限と下限の値を設定し、1700の更新ボタンを押したのと同様に設定を有効にする。妥当でない場合は、上限と下限の値は割当率範囲設定1100には反映されない。

【0179】契約入力ユーザインタフェース3400によって契約内容の変更指示が発行され、それに対して、入力ユーザインタフェース1001が顧客画面に出力する契約確認ユーザインタフェース3200は、前記第6実施形態の場合と同様である。また、契約出力ユーザインタフェース3300についても前記第6実施形態と同様である。

【0180】なお、負荷計測手段が計測した負荷または適応制御手段が決定した各論理区画に対する計算機資源の割当率とを出力する出力ユーザインタフェースを有し、該出力ユーザインタフェースは、前記負荷計測手段が計測した各L P A R上のOSの負荷と該適応制御手段が決定した各L P A Rに対する計算機資源の割当率を時系列として表示することを特徴とする仮想計算機システムであってもよい。

【0181】また、負荷計測手段が計測した負荷または該適応制御手段が決定した各論理区画に対する計算機資源の割当率とを出力する出力ユーザインタフェースを有し、適応制御手段が各L P A Rに対する計算機資源の割当率を変更すると、前記出力ユーザインタフェースは当該変更の理由を表示することを特徴とする仮想計算機システムとしてもよい。

【0182】また、物理計算機を複数のL P A Rに分割し、各L P A R上でそれぞれOSが動作させ、各L P A Rに対する物理計算機の資源の割当を制御するハイパバイザを有する仮想計算機システムであって、各L P A R上のOSの負荷を計測する負荷計測手段と、該負荷計測手段によって計測された各L P A R上のOSの負荷に基づいて、各L P A Rに割当てた計算機資源の割当率を決定し、該割当率がこれまで割り当ててきた割当率と異なる場合は、ハイパバイザに対し資源の割当率変更を指示する適応制御手段と、前記負荷計測手段が計測した負荷と適応制御手段が決定した各L P A Rに対する計算機資源の割当率を時系列としてファイル（ログファイル）に記録するログ記録手段を有し、前記ハイパバイザは該適応制御手段からの指示に従って各L P A Rに対する計算機資源の割当率を動的に変更する手段を設けたことを特徴とする仮想計算機システムとしてもよく、さらに、前記ログ記録手段は、適応制御手段が各L P A Rに対する計算機資源の割当率を変更した履歴をログファイルに記録することを特徴とする仮想計算機システムとしてもよい。

【0183】また、各顧客毎に契約条件を設定する契約ユーザインタフェースとを設け、該契約ユーザインタフェースに、顧客に割当られた論理区画上のOSの負荷や

論理区画に対する計算機資源の割当率や割当率の切り替え理由を時系列的に顧客の計算機の画面に表示する手段を設けたことを特徴とする仮想計算機システムであっても良い。

【0184】また、ユーザインタフェースには、指定された設定を設定ファイルに書き出す手段と、該設定ファイルを読みこんで該設定ファイルに保存されていた設定を該ユーザインタフェース上に復元する手段を設けたことを特徴とする仮想計算機システムとしてもよい。

【0185】今回開示した実施の形態は、全ての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味及び内容の範囲での全ての変更が含まれることが意図される。

【図面の簡単な説明】

【図1】本発明の仮想計算機を動作させる物理計算機の構成を示す図である。

【図2】同じく仮想計算機概念図である。

【図3】仮想計算機のモジュール構成を示す概略図である。

【図4】設定などを入力するための入力ユーザインタフェースの画面イメージを示す図である。

【図5】各L P A Rの負荷やC P U割当率の時系列及び割当変更理由を表示する出力ユーザインタフェースの画面イメージを示す図である。

【図6】負荷に基づいて各L P A Rに対する計算機資源の割当を変更する負荷適応制御の処理の概要を説明するフローチャートである。

【図7】負荷計測部で行われる負荷計測処理の流れを示すフローチャートである。

【図8】適応制御部が、負荷計測部で計測した負荷からC P U割当率を決定する処理の流れを示すフローチャートである。

【図9】割当率決定処理で行われる負荷に加工を施す処理の流れを示すフローチャートである。

【図10】割当率決定処理で行われる仮C P U割当率の計算処理の流れを示すフローチャートである。

【図11】同じく、仮C P U割当率を閾値法により計算する場合のフローチャートである。

【図12】割当率修正処理の一例を示すフローチャートで、その前半部である。

【図13】同じく、割当率修正処理の一例を示すフローチャートで、その中間部である。

【図14】同じく、割当率修正処理の一例を示すフローチャートで、その後半部である。

【図15】第2の実施形態を示し、仮想計算機のモジュール構成を示す概略図である。

【図16】第3の実施形態を示し、仮想計算機のモジュール構成を示す概略図である。

【図17】第4の実施形態を示し、入力ユーザインタフェースの画面イメージを示す図である。

【図18】第5の実施形態を示し、出力ユーザインタフェースをログ記録部とした場合の概念図である。

【図19】第6の実施形態を示し、仮想計算機のモジュール構成を示す概略図である。

【図20】同じく、契約入力ユーザインタフェースの画面イメージを示す図である。

【図21】同じく、契約確認ユーザインタフェースの画面イメージを示す図である。

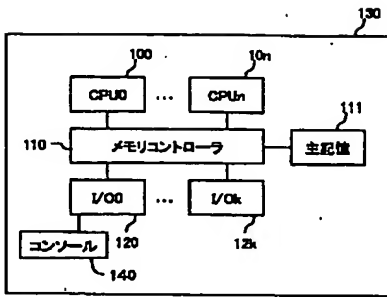
【図22】同じく、契約出力ユーザインタフェースの画面イメージを示す図である。

【図23】第7の実施形態を示し、契約入力ユーザインタフェースの画面イメージを示す図である。

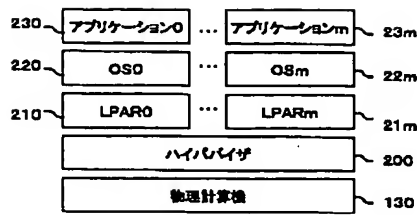
【符号の説明】

100、・・・、10m CPU0、・・・CPUm
 110 メモリコントローラ
 111 主記憶
 120、・・・、12m I/O0、・・・、I/Om
 130 物理計算機
 140 コンソール
 200 ハイパバイザ
 210、・・・、21m L P A R0、・・・、L P A Rm
 220、・・・、22m OS0、・・・、OSm
 300 適応制御部
 400、・・・、40m 負荷計測部0、・・・、負荷計測部m
 1000 ユーザインタフェース
 1001 入力ユーザインタフェース
 1002 出力ユーザインタフェース
 1004 ログ記録部
 1100 割当率範囲設定
 1200 割当率計算設定
 1300 負荷加工設定
 1400 計測負荷設定
 1500 制御インターバル設定
 1600 適応制御有効設定
 1800、・・・、180m OS0の負荷の時系列表示欄、・・・、OSmの負荷の時系列表示欄
 1810 各L P A Rに対するC P U割当率の時系列表示欄
 1820 割当変更理由表示欄
 3000、・・・、300m 契約ユーザインタフェース0、・・・、契約ユーザインタフェースm

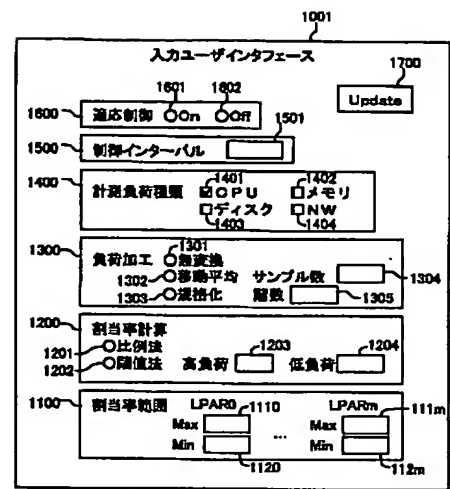
【図1】



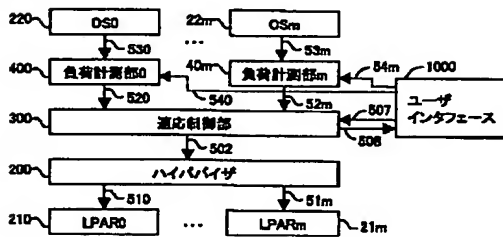
【図2】



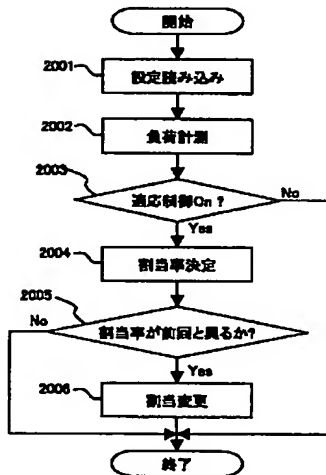
【図4】



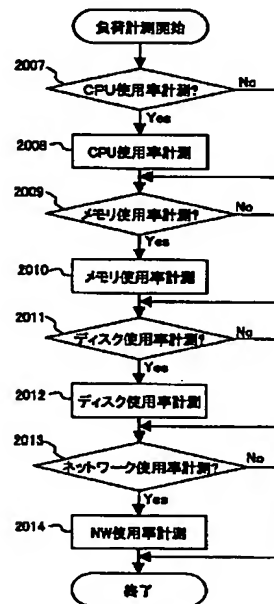
【図3】



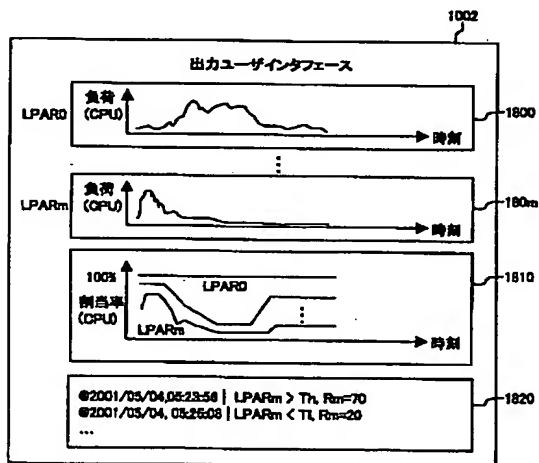
【図6】



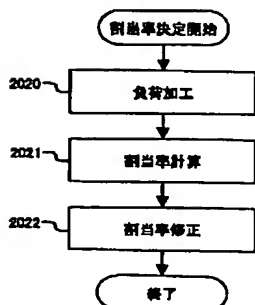
【図7】



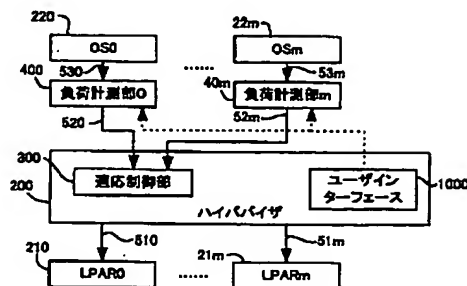
【図5】



【図8】



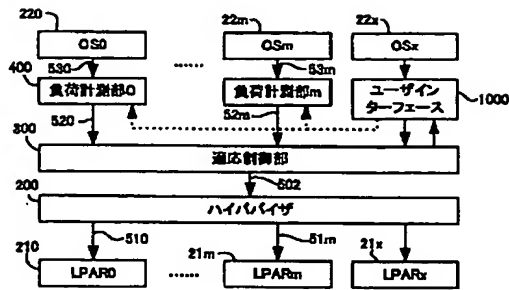
【図16】



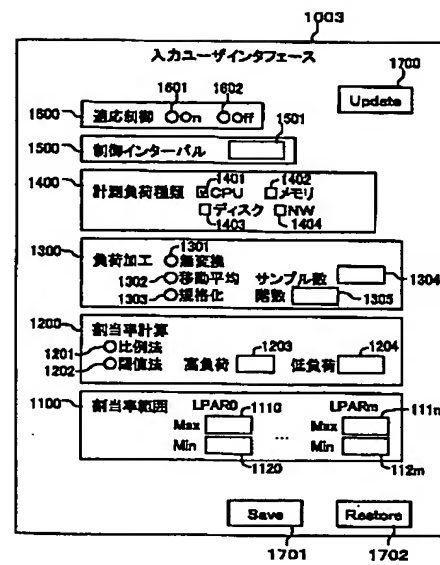
【図18】



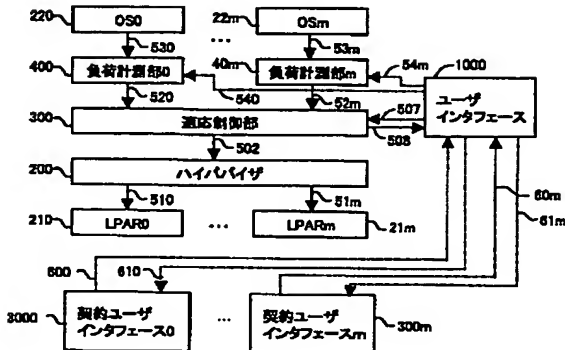
【図15】



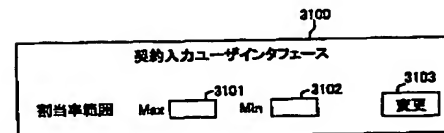
【図17】



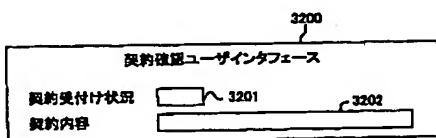
【図19】



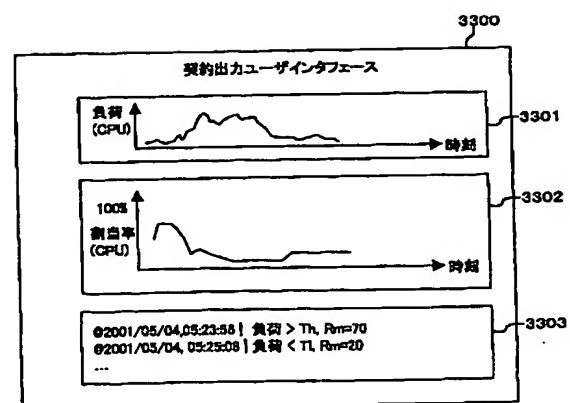
【図20】



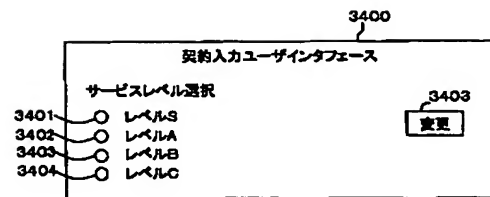
【図21】



【図22】



【図23】



フロントページの続き

(72)発明者 濱中 直樹
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内

Fターム(参考) 5B098 AA10 GA02 GC00 GC10 GD02
GD03 GD07 GD14 HH04